CLAIMS

We claim:

1   1.   A driver system interface, comprising:

2   an operating system (OS) interface to process a plurality of messages for a plurality of

3   internal driver entities ; and

4   a message controller coupled to the OS interface to transfer the plurality messages.

1   2.   The driver system interface of claim 1, further comprising:

2   a platform interface coupled to the message controller to provide platform specific

3   information to the message controller.

1   3.   The driver system interface of claim 1, wherein the message controller

2   communicates with the OS interface through functions.

1   4.   The driver system interface of claim 1, further comprising:

2   a plurality of message channels to communicate the plurality of messages to and from

3   the plurality of internal driver entities.

1   5.   The driver system interface of claim 4, the message controller comprising:

2   a plurality of installable components corresponding to the plurality of message

3   channels

1   6.   The driver system interface of claim 5, wherein the plurality of installable

2   components comprise function pointers corresponding to functions in the OS interface.

1   7.   The driver system interface of claim 1, wherein the message controller routes

2   the plurality of messages to a plurality of internal entities.

1   8.   The driver system interface of claim 1, the OS interface comprising:

2   an external interface to communicate with the plurality of external driver entities.

1   9.   The driver system interface of claim 1, wherein each message of the plurality

2   of messages comprises a message header portion containing routing information for the

3   message controller and a message information portion containing data related to an action for

4   a target entity to perform.

1   10.   The driver system interface of claim 9, wherein the message header portion

2   comprises an event variable to indicate a unique event for a corresponding message channel

3   and a message channel identifier variable to indicate the corresponding message channel.

1   11.   A communications driver comprising:

2   a network driver interface; and

3   a miniport driver coupled to the network driver interface, the miniport driver

4   comprising:

5   a system interface abstraction layer (SIAL) comprising:

6   an operating system (OS) interface to process a plurality of messages for a plurality of

7   internal driver entities; and

8   a message controller coupled to the OS interface to transfer the plurality of messages.

1   12.   The communications driver of claim 11, the SIAL further comprising:

2   a platform interface coupled to the message controller for providing platform specific

3   information and commands to the message controller.

1   13.   The communications driver of claim 11, wherein the message controller

2   communicates with the OS interface through functions.

1   14.   The communications driver of claim 11, the message controller further

2   comprising:

3   a plurality of message channels, each message channel for communicating a subset of

4   the plurality of messages to and from a corresponding subset of the plurality of internal

5   devices to a specific external device.

         15.   The communications system driver of claim 14, wherein the message

2   controller comprises a plurality of installable components corresponding to the plurality of

3   message channels.

1   16.   The communications system driver of claim 15, wherein the plurality of

2   installable components comprise function pointers corresponding to functions in the OS

3   interface.

1   17.   The communications driver of claim 11, the OS interface comprising:

2   an external interface for communicating with the plurality of external entities.

1   18.   The communications system driver of claim 11, the network driver interface

2   further comprising:

3   a dynamic messaging library coupled to the SIAL.

1   19.   The communications system driver of claim 11, wherein each message of the

2   plurality of messages comprises a message header portion containing routing information for

3   the message controller and a message information portion containing data related to an action

4   for a target entity to perform.

1   20.   The communications system driver of claim 19, wherein a message header

2   comprises an event variable to indicate a unique event for a corresponding message channel

3   and a message channel identifier variable to indicate the corresponding message channel.

1   21.   A communications card, the communications card comprising:

2   a communications system driver comprising:

3   a network driver interface;

4   a miniport driver coupled tot he network driver interface; and

5   a system interface abstraction layer (SIAL) coupled to the network

6   driver interface and the miniport driver, the SIAL comprising:

7   an operating system (OS) interface for processing a plurality

8   messages to and from a plurality of entities internal to the OS; and

9   a message controller coupled to the OS interface for translating

10   the messages and routing the message to and form an entity external to

11   the OS.

1   22.   The communications card of claim 21, the SIAL further comprising:

2   a platform interface coupled to the message controller for providing platform specific

3   information and commands to the message controller.

1   23.   The communications card of claim 21, wherein the message controller

2   communicates with the OS interface through functions.

1   24.   The communications card of claim 21, the message controller further

2   comprising:

3   a plurality of message channels, each message channel for communicating a subset of

4   the plurality of messages to and from a corresponding subset of the plurality of internal

5   devices to a specific external device.

1   25.   The communications card of claim 24, wherein a message header comprises

2   an event variable to indicate a unique event for a corresponding message channel and a

3   message channel identifier variable to indicate the corresponding message channel.

1   26.   The communications card of claim 24, wherein the message controller

2   comprises a plurality of installable components corresponding to the plurality of message

3   channels.

1   27.   The communications card of claim 26, wherein the plurality of installable

2   components comprise function pointers corresponding to functions in the OS interface.

1   28.   The communications card of claim 21, the OS interface comprising:

2   a external interface for communicating with the plurality of external entities.

1   29.   The communications card of claim 21, the communications card further

2   comprising:

3   a dynamic messaging library coupled to the SIAL.

1   30.   A communications driver, comprising:

2   a network driver interface; and

3   a driver system interface comprising:

4   an external interface to communicate with a plurality of external driver entities; and

5   an internal interface to communicate with a plurality of internal driver entities.

1   31.   The communications driver of claim 30, wherein the external interface handles

2   the semantics of the plurality of external driver entities.

1   32.   The communication driver of claim 30, wherein the external interface is a

2   portion of an operating system (OS) interface.

1   33.   The communication driver of claim 30, wherein the internal interface

2   comprises a message controller to control a plurality of message channels to pass a plurality

3   of messages between the plurality of external driver entities and the plurality of internal

4   driver entities.

1   34.   A method of abstracting a driver system interface, the method comprising the

2   steps of:

3   creating a platform specific and operating system specific message channel between

4   an internal driver entity and an external driver entity; and

5   routing a software message between the internal driver entity and the external driver

6   entity through the platform specific and operating specific message channel.

1   35.   The method of claim 34, further comprising the steps of:

2   creating a plurality of platform specific and operating specific message channels

3   between a plurality of internal driver entities and a plurality of external driver entities; and

4   routing a plurality of software messages between the plurality of internal driver

5   entities and the plurality of external driver entities.

1   36.   The method of claim 34, further comprising the steps of:

2   creating a platform specific and operating specific message channel between a first

3   internal driver entity and a second internal driver entity; and

4   routing a software message between the first driver entity and the second driver entity

5   through the platform specific and operating system specific message channel.

1   37.   The method of claim 34, wherein the routing step is performed by an

2   installable component corresponding to the message channel.

1   38.   The method of claim 34, wherein the software message comprises a header

2   portion containing routing information and an information portion containing data specific to

3   an action to be performed by a target driver entity.

1   39.   The method of claim 34, wherein the internal driver entity is a miniport driver

2   module and the external driver entity is a non-miniport driver module.